

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Applicant: Hershkovich et al.

§  
§  
§  
§  
§  
§  
§  
§  
§  
§  
§

Serial No.: 10/690,556

Filed: October 23, 2003

For: Search Method Using Coded Keys

Examiner: Jean B. Fleurantin

Group Art Unit: 2162

Attorney

Docket: 2694/24

Commissioner of Patents and Trademarks  
Alexandria, Virginia 22313FAX NO. 571-273-4035INFORMAL

Examiner Fleurantin:

Further to our conversation yesterday, I thoroughly reviewed the MPEP reference on duplicate claims that you had referred to.

Upon review of that reference, and other related references, I am even more firmly convinced that instant claims 1 and 21 are NOT duplicate claims.

As I had mentioned yesterday, claim 1 has a first, more broad aspect, and a second, more narrow aspect, with respect to claim 21:

- the more broad aspect pertains to the type of data structure: claim 1 includes non-tree structures, examples of which (along with references) are provided hereinbelow;

S/N: 10/690,556  
Art Unit: 2162

2

26-Jun-07  
Atty. Dkt.:2694/24

- the more narrow aspect pertains to the limitation that the function for performing the pre-determined transformation is substantially independent of specific content of each key entries. Claim 21 does not have this limitation.

Since claims 1 and 21 clearly and significantly differ in scope, these claims do not appear to be duplicate claims as defined in the MPEP guidelines.

I will try to reach you by phone to discuss briefly, as necessary, and would like to thank you for your courteous and devoted attention to this matter.

Yours respectfully,

Mark M. Friedman  
Attorney for Applicant  
Registration No. 33,883

Date: June 26, 2007

## **Search Data Structures which are not (necessarily) Trees**

### Reference 1:

- Section 10.2: Linked Lists.
- Sections 19.1.2, Chapter 20: Heaps, which are collections of trees.
- Part VI: Graph data structures used for search (not necessarily trees<sup>1</sup>).

### Reference 2:

- Sections 3.3-3.5: Linked Lists
- Section 3.6: Strings
- Section 3.7: Compound Data Structures<sup>2</sup>
- Section 5.8: Graph data structures<sup>3</sup>
- Chapter 14: Hash Tables
- Chapter 16: Database data structures (not necessarily trees<sup>1</sup>)

### Reference 3:

- Section 6.2: Tables (ordered and unordered and not necessarily a tree)
- Section 6.3: Trie (and NOT TREE) structures. Can be structures as a collection of trees, or as trie tables (not necessarily trees).

---

<sup>1</sup> By "not necessarily trees" we mean that such data structures can be implemented with trees, partially with trees, or without trees at all.

<sup>2</sup> By compound data structures we mean structures, which may contain for instance linked trees, or tabularly-related trees, or ... or any other data structures different from trees. In the case where the basic elements are trees, a tree may be considered as a subset of a Compound Data Structure (but not the other way around).

<sup>3</sup> Graph data structures may be (rarely) trees. More commonly, graphs are arranged as mesh structures (where multiple alternative "paths" in between two graph vertices are present - a most common situation). Such graphs may be used to implement various search operations.

## References

### **1. Introduction to Algorithms, Second Edition**

**Thomas H. Cormen**

**Charles E. Leiserson**

**Ronald L. Rivest**

**Clifford Stein**

**The MIT Press**

Cambridge , Massachusetts London, England

McGraw-Hill Book Company

Boston Burr Ridge , IL Dubuque , IA Madison , WI New York San Francisco St.  
Louis

Montréal Toronto

### **2. Algorithms in Java: Parts 1-4, Third Edition**

**By Robert Sedgewick**

Publisher: Addison Wesley

Pub Date: July 23, 2002

ISBN: 0-201-361205,

768 pages

### **3. THE ART OF COMPUTER PROGRAMMING**

**Volume 3 / Sorting and Searching**

**By Donald E. Knuth**

Publisher: Addison Wesley

Pub Date: January 2000

ISBN: 0201896850